Binary Search Tree

```
//node
Struct node {
    int key;
    Struct node *right;
    Struct node *left;
};
//Creating a node.
Struct node * Create (int data) {
    Struct node *temp;
    temp = (Struct node*) malloc (sizeof (Struct node));
    temp -> key = data;
    temp -> right = temp -> left = NULL;
    return temp;
}
Void insert (Struct node *root, Struct node *temp) {
    if (temp -> data < root -> data) {
        if (root -> left != NULL)
            insert (root -> left, temp);
        else
            root -> left = temp;
    } else {
        if (root -> right != NULL)
            insert (root -> right, temp);
        else
            root -> right = temp;
    }
}
```

```
void inorder(struct node *root){
    if(root==NULL)
        return;
    inorder(root->left);
    printf("%d", root->data);
    inorder(root->right);
}
void preorder(struct node *root){
    if(root==NULL)
        return;
    printf("%d", root->data);
    preorder(root->left);
    preorder(root->right);
}
void postorder(struct node *root){
    if(root==NULL)
        return;
    postorder(root->left);
    postorder(root->right);
    printf("%d", root->data);
}
```